

Design Pattern

Апликацијата која што ја изработивме, со тек на време сфативме дека има многу код што се повторува, на одредени делови и не е лесно читлив, па со примена на знаењата од предметот што го изучуваме се потрудивме да го подобриме.

За back-end делот го применивме MVC шаблонот. MVC шаблонот се применува многу при развивање на веб апликации со цел да се намали повторувањето на кодот, да се подобри структурата и читливоста на кодот, а и олесни одржување и додавање на нови функционалности. Со MVC овозможивме да се намали повторувањето на код со тоа што секоја компонента се обидовме да направиме да е должна за функционалности што се специфични за неа, со цел да нема преклопувања на функционалности.

Кодот го поделивме на service, model, repository,controller, за проектот да е подобро структуриран и таму ни се сместени конкретните фајлови потребни за моделот и итн... Во model пакетот се содржат класите кои ги претставуваат ентитетите врз кои ја обработуваме бизнис логиката. Во service ги имаме интерфејсите кои се користат во бизнис логиката и се користат во апликацијата, тие содржат коментари со параметри, и што точно треба да врати конкретната метода. Исто така таму се содржи пакетот implementation кои ја содржат имплементацијата на тие интерфејси. Имаме repository кое со помош на JPA комуницира со нашата база на податоци.

Апликацијата подоцна ја поделивме на повеќе независни делови, односно микросервиси, главниот дел претставува во stocksapp фолдерот, каде што најголемиот дел од апликацијата се наоѓа таму, и ги добива првичните requests кои доколку е потребно ги препраќа кон останатите микросервиси за да се обработат.

Front-end делот се обидовме да го рефакторираме со тоа што деловите во проектот се обидовме логички да ги разделиме со тоа што на ниво на проектот се одделени во различни пакети како: components,pages итн... Доколку некоја од функциите се користеше повеќе од еднаш на различни страници, ја одделувавме со цел да се реискористи поедноставно, но за среќа во front-end делот немавме многу код како функции што се употребуваат повеќе пати бидејќи секоја страница претставуваше различна логика. Деловите како footer и nav bar(лента за навигација) кои беа исти за секоја страница се одделени настрана и потоа вметнати со таг, со цел да не се повторува истиот код

Секој дел е контејнеризиран вклучувајќи ја и базата на податоци, со помош на docker, користејќи docker-compose за поедноставно стартување на сите контејнери.